Automatic Term Mismatch Diagnosis for Selective Query Expansion

Le Zhao
Language Technologies Institute
Carnegie Mellon University
Pittsburgh, PA, USA
lezhao@cs.cmu.edu

Jamie Callan
Language Technologies Institute
Carnegie Mellon University
Pittsburgh, PA, USA
callan@cs.cmu.edu

ABSTRACT
People are seldom aware that their search queries frequently mismatch a majority of the relevant documents. This may not be a big problem for topics with a large and diverse set of relevant documents, but would largely increase the chance of search failure for less popular search needs. We aim to address the mismatch problem by developing accurate and simple queries that require minimal effort to construct. This is achieved by targeting retrieval interventions at the query terms that are likely to mismatch relevant documents. For a given topic, the proportion of relevant documents that do not contain a term measures the probability for the term to mismatch relevant documents, or the term mismatch probability. Recent research demonstrates that this probability can be estimated reliably prior to retrieval. Typically, it is used in probabilistic retrieval models to provide query dependent term weights. This paper develops a new use: Automatic diagnosis of term mismatch. A search engine can use the diagnosis to suggest manual query reformulation, guide interactive query expansion, guide automatic query expansion, or motivate other responses. The research described here uses the diagnosis to guide interactive query expansion, and create Boolean conjunctive normal form (CNF) structured queries that selectively expand 'problem' query terms while leaving the rest of the query untouched. Experiments with TREC Ad-hoc and Legal Track datasets demonstrate that with high quality manual expansion, this diagnostic approach can reduce user effort by 33%, and produce simple and effective structured queries that surpass their bag of word counterparts.

Categories and Subject Descriptors
H.3.3 [**Information Search and Retrieval**]: *Query formulation, Retrieval Models*

General Terms
Algorithms, Experimentation, Theory

Keywords
Query term diagnosis, term mismatch, term expansion, Boolean conjunctive normal form queries, simulated user interactions

1. INTRODUCTION
Vocabulary mismatch between queries and documents is known to be important for full-text search. Recent research formally defined the term mismatch probability, and showed that on average a query term mismatches (fails to appear in) 40% to 50% of the documents relevant to the query [32]. With multi-word queries, the percentage of relevant documents that match the whole query can degrade very quickly. Even when search engines do not require all query terms to appear in result documents, including a query term that is likely to mismatch relevant documents can still cause the mismatch problem: The retrieval model will penalize the relevant documents that do not contain the term, and at the same time favor documents (false positives) that happen to contain the term but are irrelevant. Since the number of false positives is typically much larger than the number of relevant documents for a topic [8], these false positives can appear throughout the rank list, burying the true relevant results.

This work is concerned with the term mismatch problem, a long standing problem in retrieval. What's new here is the term level diagnosis and intervention. We use automatic predictions of the term mismatch probability [32] to proactively diagnose each query term, and to guide further interventions to directly address the problem terms. Compared to prior approaches, which typically handle the query as a whole, the targeted intervention in this work generates simple yet effective queries.

Query expansion is one of the most common methods to solve mismatch. We use the automatic term mismatch diagnosis to guide query expansion. Other forms of intervention, e.g. term removal or substitution, can also solve certain cases of mismatch, but they are not the focus of this work. We show that proper diagnosis can save expansion effort by 33%, while achieving near optimal performance.

We generate structured expansion queries of Boolean conjunctive normal form (CNF) -- a conjunction of disjunctions where each disjunction typically contains a query term and its synonyms. Carefully created CNF queries are highly effective. They can limit the effects of the expansion terms to their corresponding query term, so that while fixing the mismatched terms, the expansion query is still faithful to the semantics of the original query. We show that CNF expansion leads to more stable retrieval across different levels of expansion, minimizing problems such as topic drift even with skewed expansion of part of the query. It outperforms bag of word expansion given the same set of high quality expansion terms.

2. RELATED WORK
This section discusses how this work relates to the other research that tries to solve the mismatch problem. In particular, research on predicting term mismatch and on conjunctive normal form (CNF) structured queries forms the basis of this work.

2.1 Term mismatch and automatic diagnosis
Furnas et al. [7] were probably the first to study *vocabulary mismatch* quantitatively, by measuring how people name the same

SIGIR'12, August 12–16, 2012, Portland, Oregon, USA.

concept/activity differently. They showed that on average 80-90% of the times, two people will name the same item differently. The best term only covers about 15-35% of all the occurrences of the item, and the 3 best terms together only cover 37-67% of the cases. Even with 15 aliases, only 60-80% coverage is achieved. The authors suggested one solution to be "unlimited aliasing", which led to the Latent Semantic Analysis (LSA) [6] line of research.

Zhao and Callan [32] formally defined the *term mismatch probability* to be $P(\bar{t} \mid R)$, the likelihood that term *t* does not appear in a document *d*, given that *d* is relevant to the topic ($d \in R$), or equivalently, the proportion of relevant documents that do not contain term *t*. Furnas et al. [7]'s definition of vocabulary mismatch is query independent, and can be reduced to an average case of Zhao and Callan [32]'s query dependent definition.

The complement of term mismatch is the term recall probability: $P(t \mid R)$. A low $P(t \mid R)$ means term *t* tends not to appear in the documents relevant to the topic. This query dependent probability $P(t \mid R)$ is not new in retrieval research. It is known to be part of the Binary Independence Model (BIM) [23], as part of the optimal term weight. Accurate estimation of $P(t \mid R)$ requires knowledge of *R* -- the relevant set of a topic, which defeats the purpose of retrieval, and $P(t \mid R)$ was thought to be difficult to estimate.

Recent research showed that $P(t \mid R)$ can be reliably predicted without using relevance information of the test topics [8,20,32]. Zhao and Callan [32] achieved the best predictions from being the first to design and use query dependent features for prediction, features such as term centrality, replaceability and abstractness.

Previously, $P(t \mid R)$ predictions were used to adjust query term weights of inverse document frequency (idf)-based retrieval models such as Okapi BM25 and statistical language models. Term weighting is not a new technique in retrieval research, neither is predicting term weights.

Our work is a significant departure from the prior research that predicted $P(t \mid R)$. We apply the $P(t \mid R)$ predictions in a completely new way, to automatically diagnose term mismatch problems and inform further interventions.

2.2 CNF structured expansion

Query expansion is one of the most common ways to solve mismatch. In recent years, the research community has focused on expansion of the whole query, for example using pseudo relevance feedback [17]. This form of expansion is simple to manage and effective. It also allows introduction of expansion terms that are related to the query as a whole, even if their relationship to any specific original query term is tenuous.

When *people* search for information, they typically develop queries in Boolean conjunctive normal form (CNF). CNF queries are used by librarians [16,12], lawyers [2,26] and other expert searchers [4,13,21]. Each conjunct represents a high-level concept, and each disjunct represents alternate forms of the concept. Query expansion is accomplished by adding disjuncts that cover as many ways of expressing the concept as possible.

For example, the query below from TREC 2006 Legal Track [2]

"*sales of tobacco to children*"

is expanded manually into

(*sales* OR *sell* OR *sold*) AND
(*tobacco* OR *cigar* OR *cigarettes*) AND
(*children* OR *child* OR *teen* OR *juvenile* OR *kid* OR *adolescent*)

CNF queries ensure precision by specifying a set of concepts that must appear (AND), and improve recall by expanding alternative forms of each concept. Compared to LSA or bag of word expansion, CNF queries offer control over *what query terms to expand* (*the query term dimension*) and *what expansion terms to use for a query term* (*the expansion dimension*).

However, these two dimensions of flexibility also make automatic formulation of CNF queries computationally challenging, and makes manual creation of CNF queries tedious. The few experiments demonstrating effective CNF expansion either used manually created queries or only worked for a special task. Hearst [13] and Mitra et al. [21] used ranked Boolean retrieval on manual CNF queries. Zhao and Callan [31] automatically created CNF queries for the question answering task, based on the semantic structure of the question.

Along the two directions of term diagnosis and expansion, prior research has focused on identifying synonyms of query terms, i.e. the expansion dimension. Google has patents [15] using query logs to identify possible synonyms for query terms in the context of the query. Jones and colleagues [14] also extracted synonyms of query terms from query logs. They called it query substitutions. Wang and Zhai [28] mined effective query reformulations from query logs. Dang and Croft [5] did the same with TREC Web collections. Xue, Croft and Smith [30] weighted and combined automatic whole-query reformulations, similar to the way alternative structured queries were combined in [31]. If more extensive expansions were used, the more compact CNF expansion would be a reasonable next step. Because of reasons such as suboptimal quality of expansions or insufficient number of topics for evaluation, prior research on ad hoc retrieval has not seen automatic CNF expansion to outperform keyword retrieval. Perhaps the only exception is the related problem of context sensitive stemming [27,22,3], where expansion terms are just morphological variants of the query terms, which are easier to identify and more accurate (less likely to introduce false positives).

Such prior work tried to expand any query term, and did not exploit the term diagnosis dimension, thus they essentially expanded the query terms whose synonyms are easy to find.

This work focuses on selectively expanding the query terms that really need expansion, a less well studied dimension. Exploiting this diagnosis dimension can guide further retrieval interventions such as automatic query reformulation or user interaction to the areas of the query that need help, leading to potentially more effective retrieval interventions. It also reduces the complexity of formulating CNF queries, manually or automatically. The prior research on synonym extraction is orthogonal to this work, and can be applied with term diagnosis in a real-world search system.

2.3 Simulated interactive expansions

Our diagnostic intervention framework is general and can be applied to both automatic and manual expansions. However, our experiments are still constrained by the availability of effective intervention methods. That is why we use manual CNF expansion, which is highly effective. To avoid using the expensive and less controllable online user studies, we use existing user-created CNF queries to simulate online diagnostic expansion interactions.

Simulations of user interactions are not new in interactive retrieval experiments. Harman [9] simulated relevance feedback experiments by using relevance judgements of the top 10 results, and evaluated feedback retrieval on the rest of the results. White et al. [29] also used relevance judgements and assumed several user browsing strategies to simulate users' interactions with the search interface. The interaction simulations were used as user feedback to select expansion terms. Similarly, Lin and Smucker [18] assumed a set of strategies that define how the user browses

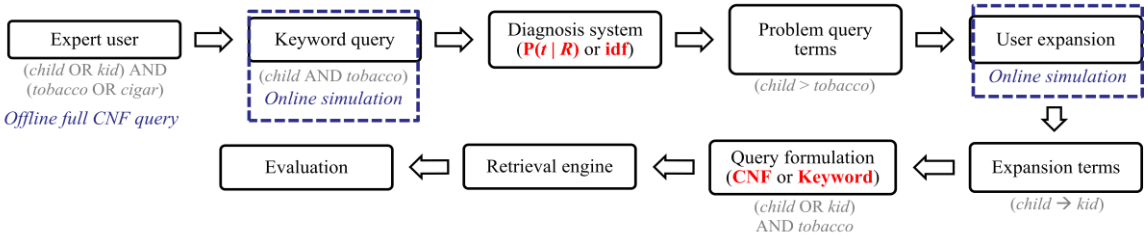

Figure 1. Simulated diagnostic expansion, with query examples in gray, simulated steps in dashed boxes and methods to test in bold red font.

the search results, to simulate and evaluate a result browsing interface using known relevance judgements.

Compared to the prior work, our simulations never explicitly use any relevance judgements, and only make a few relatively weak assumptions about the user. We simulate based on existing user created Boolean CNF queries, which can be seen as recorded summaries of real user interactions. These fully expanded queries are used to simulate selective expansion interactions.

3. DIAGNOSTIC INTERVENTION

This section discusses in more detail the diagnostic intervention framework, and shows how term diagnosis can be applied and evaluated in end-to-end retrieval experiments in an ideal setting.

We hope to answer the following questions. Suppose the user is willing to invest some extra time for each query, how much effort is needed to improve the initial query (in expansion effort, how many query terms need to be expanded, and how many expansion terms per query term are needed)? When is the best performance achieved? Can we direct the user to a subset of the query terms so that less effort is needed to achieve a near optimal performance? What's an effective criterion for term diagnosis?

3.1 Diagnostic intervention framework

The diagnostic intervention framework is designed as follows. The user issues an initial keyword query. Given the query, the system selects a subset of the query terms and asks the user to fix (e.g. expand) them. The performance after user intervention is used to compare the different diagnosis strategies.

This evaluation framework needs to control two dimensions, *the diagnosis dimension* (selecting the set of problem query terms) and *the intervention dimension* (determining the amount of intervention for each selected term). Diagnosis of terms with mismatch problems can be achieved using criteria such as low predicted $P(t \mid R)$ or high idf. The intervention dimension when implemented as query expansion can be controlled by asking the user to provide a certain number of expansion terms.

3.2 Query term diagnosis methods

We consider two term diagnosis methods, idf based term diagnosis and predicted $P(t \mid R)$ based diagnosis.

The idf based diagnosis selects the query terms that have the highest idf first. Idf is known to have a correlation with $P(t \mid R)$ [8] and has been used as a feature for predicting $P(t \mid R)$ [8,20,32]. A rare term (high idf) usually means a high likelihood of mismatch, while a frequent word (e.g. stopword) would have a high $P(t \mid R)$.

Diagnosis based on predicted $P(t \mid R)$ selects the query terms with the lowest predicted $P(t \mid R)$ first. We use the best known method to predict $P(t \mid R)$ [32], which was the first to use query dependent features for prediction. It used top ranked documents from an initial retrieval to automatically extract query dependent synonyms. These synonyms were used to create some of the effective query dependent features, e.g. how often synonyms of a query term appear in top ranked documents from the initial retrieval, and how often such synonyms appear in place of the original query term in collection documents. Section 4 describes implementation details.

3.3 Possible confounding factors

To exactly follow this ideal framework, for each query, many user interaction experiments are needed – one experiment for each possible diagnostic intervention setup, preferably, one user per setup. Many factors need to be controlled, such as users' prior knowledge of the topic, the quality of the manual interventions, users' interaction time and interaction method (whether retrieval results are examined), so that the final retrieval performance will reflect the effect of the diagnostic component instead of random variation in the experiment setup. These factors are difficult to eliminate even with hundreds of experiments per topic. We show how simulations may help in the section below.

4. EXPERIMENTAL METHODOLOGY

In this section, we design the retrieval experiments to evaluate diagnostic interventions. We explain how user simulations may be an appropriate substitute for costly online experiments with human subjects. We explain how to design the diagnostic intervention experiment so that it measures the effects of term diagnosis, minimizing effects from confounding factors such as the quality of the post-diagnosis interventions. We examine the datasets used for this simulation, in particular how well the manual CNF queries fit the simulation assumptions. We also describe the whole evaluation procedure, including the implementation of the mismatch diagnosis methods which (together with the user intervention) produce the post-intervention queries, the retrieval model behind query execution, and the evaluation metrics used to measure retrieval effectiveness.

We focus on *interactive expansion* as the intervention method, using existing CNF queries to simulate interactive expansion. This is due to both the effectiveness of query expansion to solve mismatch and the lack of user data for other types of interventions.

4.1 Simulated user interactions

As explained in Section 3.3, a large number of experiments and users are needed to evaluate diagnostic expansion using online user studies. To avoid this, we use offline automatic simulations, sketched in Figure 1. There are three players in the simulation, the *user*, the *diagnostic expansion retrieval* system and the *simulation based evaluation* system. Before evaluation, the user creates fully expanded CNF queries. These manually created CNF queries are used by the evaluation system to simulate selective expansions, and are accessible to the diagnostic retrieval system only through the simulation system. During evaluation, the simulation system first extracts a basic no-expansion keyword query from the CNF query, feeding the keyword query to the diagnosis system. The diagnostic expansion system automatically diagnoses which keywords are more problematic. Then, the simulation system takes the top several problem terms, and

extracts a certain number of expansion terms for each selected query term from its corresponding conjunct in the manual CNF query. This step simulates a user expanding the selected query terms. The number of problem query terms to expand and the number of expansion terms to include are controlled by the simulation system, which will evaluate retrieval at five different selection levels and five different expansion levels. Finally, given these expansion terms for each selected query term, the diagnostic expansion system forms an expansion query and does retrieval.

For example, based on the CNF query in Section 2.2, the diagnosis method is given the keyword query *sales tobacco children*. It may see *children* as more problematic than the other terms, then a full expansion of this problem term would produce the query *sales* AND *tobacco* AND (*children* OR *child* OR *teen* OR *juvenile* OR *kid* OR *adolescent*), whose retrieval performance is evaluated as the end result of diagnostic expansion. If the evaluation system selects two query terms *sales* and *children* for expansion, with a maximum of one expansion term each, the final query would be (*sales* OR *sell*) AND *tobacco* AND (*children* OR *child*). These diagnostic expansion queries are partial expansions simulated using the fully expanded queries created by real users.

Our simulation allows us to answer the same set of questions about the diagnostic expansion system which we hope to answer through online user interactions, and requires simpler experiments. In our simulations, the same set of expansion terms is always used for a given query term, those from its corresponding CNF conjunct. Doing so minimizes the variation from the expansion terms as we measure the effects of the diagnosis component. The order in which expansion terms are added for a query term is also fixed, in the same order as they appear in the CNF conjunct. This way, we can tweak the level of expansion by gradually including more expansion terms from the lists of expansion terms, and answer how much expansion is needed for optimal performance.

Our simulation makes three assumptions about the user expansion process. We examine them below.

Expansion term independence assumption: Expansion terms from fully expanded queries are held back from the query to simulate the selective and partial expansion of query terms. This simulation is based on the assumption that the user (a random process that generates expansion terms) will produce the same set of expansion terms for a query term whenever asked to expand any subset of the query terms. Equivalently, given the topic, the expansion process for one query term does not depend on the expansion of other query terms. In reality, a human user focusing on a subset of the query terms can typically achieve higher quality expansion. Thus, selective expansion may actually do better than the reported performance from the simulations.

***Expansion term sequence assumption*:** Controlling to include only the first few expansion terms of a query term simulates and measures a user's expansion effort for that query term. It is assumed that the user would come up with the same sequence of expansion terms for each query term, no matter whether the user is asked to expand a subset or all of the query terms. A downside of this simulation is that we do not know exactly how much time and effort the user has spent on each expansion term.

***CNF keyword-query induction assumption*:** Instead of actually asking users to expand their initial queries, preexisting fully expanded CNF style queries are used to infer the original keyword query and to simulate the expansion process. For example, given the CNF query in Section 2.2, the original keyword query is assumed to be (*sales tobacco children*). This simulation assumes that the original keyword query can be reconstructed from the manual CNF query, which could be missing some original query terms (*of* and *to* in the example) or introduce new terms into the original keyword query. However, as long as we use highly effective CNF queries, it is safe to use the CNF induced keyword queries as the no-expansion baseline.

We also made an effort to ensure that our 'reverse-engineered' keyword query is faithful to the vocabulary of the original query. Given the TREC official topic description of a topic, we try to use the first term from each conjunct that appears in this description to reconstruct the keyword query. For conjuncts that do not have description terms, the first term in the conjunct is used.

For example, the topic described as *sales of tobacco to children*, with CNF query (*sales* OR *sell* OR *sold*) AND (*tobacco* OR *cigar* OR *cigarettes*) AND (*children* OR *child* OR *teen* OR *juvenile* OR *kid* OR *adolescent*), would have (*sales tobacco children*) as the unexpanded keyword query. If the description were *sell tobacco to children*, the keyword query would be instead (*sell tobacco children*), even when *sales* appears first in its conjunct.

4.2 Effects of confounding factors

Using user simulations instead of real users can eliminate confounding factors such as the user's prior knowledge of the topic and other details of the user interaction process.

This work tests the hypothesis that term diagnosis can effectively guide query expansion. However, two factors directly determine the end performance of diagnostic expansion, 1) the effectiveness of term diagnosis, and 2) the benefit from expansion.

Since our focus is on diagnosis, not query expansion, one of the most important confounding factors is the quality of the expansion terms, which we leave out of the evaluation by using a fixed set of high quality expansion terms from manual CNF queries to simulate an expert user doing manual expansion.

Automatic query expansion is more desirable in a deployed system, but the uncertain quality of the expansion terms can confuse the evaluation. Thus, it is not considered in this paper.

4.3 Datasets and manual CNF queries

Datasets with high quality manual CNF queries are selected to simulate and evaluate diagnostic expansion. Four different datasets have been used, those from TREC 2006 and 2007 Legal tracks, and those from TREC 3 and 4 Ad hoc tracks. They are selected in pairs, because training data is needed to train the $P(t \mid R)$ prediction model. Here, the TREC 2006 (39 topics) and TREC 3 (50 topics) datasets are used for training the baseline model parameters and the $P(t \mid R)$ prediction models, while TREC 2007 (43 topics) and TREC 4 (50 topics) are used for testing.

4.3.1 TREC Legal track datasets

The TREC Legal tracks contain Boolean CNF queries created through expert user interaction. They are fairly specific, averaging 3 conjuncts per topic, i.e., 3 concepts conjoined to form a query. The information needs of the Legal track topics are fictional, but mimic the real cases.

The lawyers who created the TREC Legal queries know what the collection is, and have expert knowledge of what terminology the corpus documents might use to refer to a concept being requested. The lawyers would give very high priority to the recall of the queries they create. They tried to fully expand every query term, so as not to miss any potentially relevant document. An effort to avoid over-generalizing the topic was also made. However, the lawyers never looked at the retrieval results when creating these CNF style queries. We call this a case of *blind user interaction*, because no corpus information is accessed during user

interaction. We use the Boolean queries from [33], which achieved near best performance in TREC 2007 Legal track.

The 2006 and 2007 TREC Legal tracks share the same collection of documents. These are about 6.7 million tobacco company documents made public through litigation. They are on average 1362 words long. Many of them are OCR text, and contain spelling and spacing errors.

For relevance judgments, because of the relatively large size of the collection, a sampled pooling strategy was adopted in Legal 2007, with 555.7 judgments per topic and 101 judged relevant documents per topic.

More details about the dataset, the information needs, query formulation procedure, and relevance judgments can be found in the overview papers [2,26].

4.3.2 TREC Ad hoc track datasets

For the TREC 3 and 4 Ad hoc track datasets, high quality manual CNF queries were created by the University of Waterloo group [4]. An example query is (*responsibility* OR *standard* OR *train* OR *monitoring* OR *quality*) AND (*children* OR *child* OR *baby* OR *infant*) AND "*au pair*", where the "*au pair*" is a phrase. The information needs for the TREC 3 and 4 Ad hoc tracks are simpler (or broader), averaging 2 conjuncts per topic.

The Waterloo queries were created for the MultiText system by an Iterative Searching and Judging (ISJ) process. These queries were manually formulated with access to the results returned by the retrieval system, thesaurus and other external resources of knowledge. This constitutes a case of *user* and *corpus interaction*. Quality of the manual Boolean queries is ensured by examining the retrieval results, thus, should be better than those created from blind user interaction of the Legal tracks. Since the interaction with search results, expansion processes of the query terms may not be independent of each other. For example, in order to discover the expansion term of a query term, one may need to expand another query term first, to bring up a result document that contains the expansion term. Thus, the expansion independence assumption (of Section 4.1) is more likely to be violated by the ISJ queries than by the Legal ones.

The TREC 3 and 4 Ad hoc tracks used different collections, but they both consisted of newswire texts published before 1995. Each collection has about 0.56 million documents. The texts are non-OCR, thus cleaner than the Legal documents.

The relevance judgments of the Ad hoc tracks are deeper, because the collections are much smaller. The TREC 4 Ad hoc track made 1741 judgments per topic with 130 relevant. More details can be found in [10,11].

For all documents and queries, the Krovetz stemmer was used (more conservative than Porter), and no stopwords were removed.

4.4 Term diagnosis implementation

We explain the implementation of the diagnosis methods, idf and predicted P(t | R), in more detail.

Idf is calculated as $\log((N - \mathrm{df})/\mathrm{df})$, where N is the total number of documents in the collection and df is the document frequency of the term. This follows the RSJ formulation [23].

For P(t | R) prediction, we closely follow [32]'s method. Automatic features used for prediction include idf, and the three features derived from applying latent semantic analysis (LSA) [6] over the top ranked documents of an initial keyword retrieval.[1]

[1] Recently a more efficient method of predicting P(t | R) was developed that eliminates the need for an initial retrieval [Zhao, personal communication].

For training purposes, P(t | R) truth is calculated as $(r + 1)/(|R| + 2)$, where r is the number of relevant documents containing t and $|R|$ the total number of relevant documents for the query, with Laplace smoothing used. Support Vector Regression with RBF kernel is used to learn the prediction model.

There are 3 parameters: The number of top ranked documents for LSA, which is set at 180 for the Ad hoc datasets and 200 for the Legal track datasets, based on a monotonic relationship between this parameter and the total number of collection documents observed [32]. The number of latent dimensions to keep is fixed at 150, and the gamma parameter which controls the width of the RBF kernel is fixed at 1.5 (as in [32]).

[32] also used a feature that indicated whether a word in the query appears as a leaf node in the dependency parse of the query. Here, the feature is assumed to be 0 for all query terms, because the unexpanded query is usually not a natural language sentence or phrase, hence parsing may be inappropriate.

A small number of the original terms in these CNF queries are phrases, windowed occurrences or other complex structures. They are assumed to have a P(t | R) value of 0.5. The LSA component of the Lemur Toolkit is not designed to handle these complex terms, preventing the use of [32]'s model. This is a small handicap to our P(t | R) prediction implementation, but not to the idf method, which is based on accurate df values calculated by the Indri search engine.

4.5 The retrieval model

To achieve a state-of-the-art performance, the retrieval model needs to rank collection documents using the Boolean CNF queries. Before the mid 1990's, unranked Boolean was popular. Later research found ranked keyword to be more effective. However, to be fair, a *ranked* Boolean (e.g. soft or probabilistic) model should be used to compare with other ranking approaches.

This work adopts the language model framework, using probabilistic Boolean query execution (with Lemur/Indri version 4.10) [19]. The Boolean OR operator is still the hard OR, treating all the synonyms as if they are the same term for counting term- and document-frequencies (i.e. #syn operator in Indri query language). The Boolean AND is implemented as the probabilistic AND (the Indri #combine operator) to produce a ranking score.

Equations (1, 2) show how the retrieval model scores document d with query (a OR b) AND (c OR e). tf(a, d) is the number of times term a appears in document d. μ is the parameter for Dirichlet smoothing, which is set at 900 for the Ad hoc datasets and 1000 for the Legal datasets based on training.

Score((a OR b) AND (c OR e), d) (1)
= P((a OR b) AND (c OR e) | d)
= P((a OR b) | d) * P((c OR e) | d)

P((a OR b) | d) (2)
= (tf(a, d) + tf(b, d) + μ * (P(a | C) + P(b | C))) / (length(d) + μ)
= P(a | d) + P(b | d) (under Dirichlet smoothing)

This language model based ranked Boolean model is not the only possibility. Other ranked Boolean models include using the Boolean query as a two-tiered filter for the keyword rank list [13] [31], using the Okapi BM25 model for the conjunction [25], using probabilistic OR for the expansion terms (in Indri query language, #or instead of #syn), or using the p-norm Boolean ranking model [24]. We have tried some basic variations of the language model ranked Boolean model. Our pilot study shows that for our datasets, tiered filtering is sometimes worse than probabilistic Boolean, mostly because of the inferior ranking of the keyword

queries. Probabilistic OR (#or) is numerically similar to treating all expansion terms the same as the original term (#syn), and the two methods perform similarly in retrieval. We did not try Okapi or p-norm, because the focus of this paper is P(t | R) based diagnostic expansion, not to find the best ranked Boolean model. What is needed is one ranked Boolean model that works.

4.6 Evaluation measures

We use standard TREC evaluation measures for the datasets. Traditionally, pooled judgments and precision at certain cutoffs have been used in TREC. *Mean Average Precision (MAP)* at top 1000 is a summary statistic that cares about both top precision and precision at high recall levels, and has been used as the standard measure in TREC Ad hoc and Legal tracks.

The *statAP* measure [1] is the standard measure for TREC Legal 2007. StatAP is an unbiased statistical estimate of MAP designed to work with sampled pooling. It is unbiased in the sense that if all pooled documents were judged, the MAP value would have been the same as the mean of the estimated statAP. In traditional TREC pooling, the top 50 to top 100 documents from each submitted rank list are pooled, and all pooled documents are judged. In sampled pooling, only a sampled subset of the pool is judged. The idea is to use importance sampling to judge fewer documents while maintaining a reliable estimate of MAP. Highly ranked documents from multiple pooled submissions are more likely to be relevant, and they are sampled more by importance sampling. StatAP takes into account these sampling probabilities of the judged relevant documents, so that during evaluation, a judged relevant document with sampling probability p would be counted as a total of $1/p$ relevant documents. This is because on average $1/p - 1$ relevant documents are missed during the sampling procedure, and they are being represented by that one sampled relevant document.

For topics where some relevant documents have low sampling probabilities, statAP estimates can deviate from the true AP a lot, but according to [26], when averaged over more than 20 topics, statAP provides a reliable estimate.

5. EXPERIMENTS

These experiments test two main hypotheses. ***H1***: Mismatch diagnosis can direct expansion to the query terms that need expansion. ***H2***: Boolean CNF expansion is more effective than bag of word expansion with the same set of high quality expansion terms. To test H1, the *first* experiment verifies the accuracy of idf and P(t | R)-prediction based term diagnosis against the true P(t | R). The *second* experiment shows the effects of diagnosis by evaluating overall retrieval performance along the query term dimension (5 diagnostic selection levels) and the expansion dimension (5 expansion levels). The *third* experiment compares predicted P(t | R) diagnosis with idf based diagnosis. H2 is tested by the *fourth* experiment comparing CNF and bag-of-word expansion at various levels of diagnostic expansion.

5.1 Baseline – no expansion

Listed below is the retrieval performance of the no expansion keyword retrieval baseline on the two test sets.

Dataset	Legal 2007 (MAP/statAP)	TREC 4 (MAP)
no expansion	0.0663/0.0160	0.1973

5.2 Mismatch diagnosis accuracy

Our goal is to use idf or P(t | R) predictions to diagnose query terms, to rank them in a priority order for the user to fix (expand).

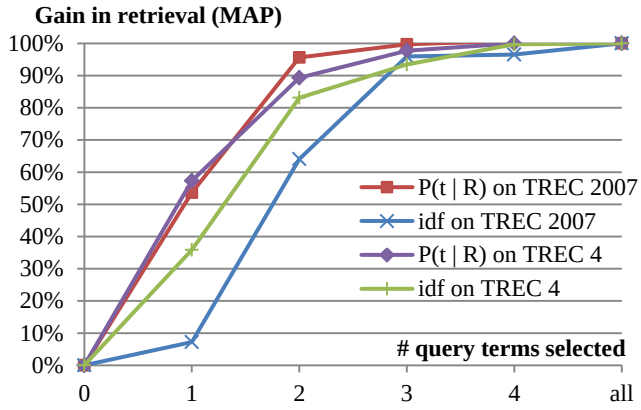

Figure 2. Relative retrieval performance gains of diagnostic expansion as the number of query terms selected for expansion increases. Calculated based on the last row of Tables 2 and 3.

This section is a unit test of the diagnosis component, in which accuracy is measured by how well the diagnosis method identifies the most problematic query terms (those most likely to mismatch). We measure how well the priority order (e.g. ascending predicted P(t | R)) ranks the query term with the true lowest P(t | R), thus use Mean Reciprocal Rank (MRR) as the measure. Rank correlation measures do not distinguish ranking differences at the top vs. bottom of the rank lists, thus are less appropriate here.

On the Legal 2007 dataset, predicted P(t | R) achieves an MRR of 0.6850, significantly higher than the MRR of 0.5523 of the idf method, significant at $p < 0.026$ by the two tailed t-test. The idf method is still much better than random chance which has an MRR of 0.383, given the average 3 conjuncts per topic.

This result that P(t | R) prediction using [32]'s method is better than idf, and idf is better than random is consistent with prior research that predicted P(t | R) [8,32].

5.3 Diagnostic expansion retrieval results

Tables 1 2 and 3 report the expansion retrieval performance of predicted-P(t | R) based and idf based diagnostic expansion, following the evaluation procedure detailed in Section 4.1. The results are arranged along two dimensions of user effort, the number of query terms selected for expansion, and the maximum number of expansion terms to include for a selected query term.

For example, results reported in column 2 row 2 selects 1 original query term of the highest idf for expansion, and a maximum of 1 expansion term is included for the selected query term. When the manual CNF query doesn't expand the selected query term, no expansion term will be included in the final query.

We are most concerned with the performance changes along each row of the tables, which are caused by the diagnosis methods. In Figure 2, we compare the relative performance gains of the different diagnosis methods as more query terms are being selected for expansion. Results based on the last row of Tables 2 and 3 are presented in Figure 2. No expansion is 0%, and full expansion of all query terms gets 100%. With only 2 query terms selected for expansion, predicted P(t | R) diagnosis is achieving 95% or 90% of the total gains of CNF expansion. Idf diagnosis is only achieving 64% or 83% of the total gains with 2 query terms, and need to fully expand 3 query terms to reach a performance close to the best (full expansion of all query terms). Thus, *predicted P(t | R) based diagnosis saves 1/3 of users' expansion effort while still achieving near optimal retrieval performance*.

Table 1. Retrieval performance of the two selective CNF expansion methods on TREC 2007 Legal track, as measured by *MAP*. The baseline unexpanded queries produced an MAP of 0.0663.

\ # terms selected	1		2		3		4		All
# expansions per term	idf	P(t | R)	idf	P(t | R)	idf	P(t | R)	idf	P(t | R)	(same)
1	0.0722	0.0778*	0.0802*	0.0825**	0.0892**	0.0896***	0.0893**	0.0904***	0.0901***
2	0.0780*	0.0805**	0.0825*	0.0921***	0.0916***	0.0938$^{\#}$	0.0947***	0.0961$^{\#}$	0.0971$^{\#}$
3	0.0766	0.0806**	0.0844**	0.0927***	0.0938***	0.0965$^{\#}$	0.0969***	0.0988$^{\#}$	0.0997$^{\#}$
4	0.0770	0.0809**	0.0859**	0.0948$^{\#}$	0.0968$^{\#}$	0.0993$^{\#}$	0.0996$^{\#}$	0.1015$^{\#}$	0.1024$^{\#}$
All	0.0754	0.0798*	0.0862**	0.0958***	0.0986$^{\#}$	0.1008$^{\#}$	0.1016$^{\#}$	0.1031$^{\#}$	0.1039$^{\#}$

* significantly better than the no expansion baseline by both randomization & sign tests at p<0.05.
** p<0.01 by both tests, *** p<0.001 by both, # p<0.0001 by both tests. (Same notation is used for the other tables.)

Table 2. Retrieval performance of the two selective CNF expansion methods on TREC 2007 Legal track, as measured by the TREC standard *statAP* measure. The baseline unexpanded queries produced a statAP of 0.0160. (Statistical significance tests are omitted, as they are inappropriate for the sampling based statAP measure [26].)

\ # query terms selected	1		2		3		4		All
# expansions per query term	idf	P(t | R)	idf	P(t | R)	idf	P(t | R)	idf	P(t | R)	(same)
1	0.0164	0.0233	0.0184	0.0246	0.0279	0.0282	0.0279	0.0282	0.0282
2	0.0176	0.0255	0.0189	0.0289	0.0273	0.0295	0.0288	0.0302	0.0300
3	0.0175	0.0256	0.0191	0.0290	0.0280	0.0304	0.0291	0.0307	0.0304
4	0.0176	0.0256	0.0194	0.0295	0.0292	0.0311	0.0301	0.0317	0.0314
All	0.0185	0.0345	0.0381	0.0490	0.0491	0.0504	0.0493	0.0508	0.0505

Table 3. Retrieval performance of the two selective CNF expansion methods on TREC 4 Ad hoc track, as measured by the TREC standard *MAP* measure. The baseline unexpanded queries produced an MAP of 0.1973.

\ # terms selected	1		2		3		4		All
# expansions per term	idf	P(t | R)	idf	P(t | R)	idf	P(t | R)	idf	P(t | R)	(same)
1	0.2087	0.2279***	0.2341*	0.2366**	0.2350**	0.2358**	0.2356**	0.2358**	0.2358**
2	0.2135	0.2392$^{\#}$	0.2503**	0.2541***	0.2552***	0.2567***	0.2578***	0.2581***	0.2581***
3	0.2187*	0.2435***	0.2538***	0.2539$^{\#}$	0.2589***	0.2608$^{\#}$	0.2619$^{\#}$	0.2622$^{\#}$	0.2622$^{\#}$
4	0.2242*	0.2489$^{\#}$	0.2654***	0.2659$^{\#}$	0.2706***	0.2731$^{\#}$	0.2753$^{\#}$	0.2756$^{\#}$	0.2756$^{\#}$
All	0.2319**	0.2526***	0.2775$^{\#}$	0.2835$^{\#}$	0.2875***	0.2916$^{\#}$	0.2935$^{\#}$	0.2938$^{\#}$	0.2938$^{\#}$

Tables 1, 2 and 3 show that the more query terms selected for expansion, the better the performance. This is not surprising, as we are using carefully expanded manual queries. Similarly, including more expansion terms (along each column) almost always improves retrieval, except for the idf method in Table 1 with only one query term selected for expansion.

The improvement over the no expansion baseline becomes significant after expanding two query terms for the idf method, and after only expanding one query term for predicted P(t | R).

Overall, P(t | R) diagnostic expansion is more stable than the idf method. This shows up in several areas. 1) Including more expansion terms always improves performance, even when only one original query term is selected for expansion. 2) Performance improvement over the no expansion baseline is significant even when only including one expansion term for one query term. These are not true for idf diagnosis. 3) Only two query terms need to be selected for expansion to achieve a performance close to the best, 33% less user effort than that of idf based diagnosis.

The statAP measure from Table 2 correlates with the MAP measure, however, the sudden increases in statAP from the 2nd last row to the last row are not present in the case of MAP. A bit of investigation shows that 1 to 2 topics benefited a lot from the extra (more than 4) expansion terms. The benefit is because of the successful matching of some relevant documents with low sampling probability, which increases statAP a lot, but not MAP. On the Legal 2007 dataset, the topic that benefited most from the more than 4 expansion terms is a topic about James Bond movies. Certainly, there are more than 4 popular James Bond movies.

Overall, predicted P(t | R) can effectively guide user expansion to improve retrieval. Expanding the first few query terms can result in significant gains in retrieval. The gain diminishes as more query terms are expanded, eventually leading to the best performance of expanding every query term.

5.4 P(t | R) vs. idf based diagnostic expansion

The subsection above shows that diagnosis can help reduce expansion effort, and that P(t | R) diagnosis results in more stable retrieval than idf. This section directly compares two retrieval experiments using predicted P(t | R) vs. idf guided expansion.

The two experiments both select 1 query term for full expansion (Table 1 last row, 2nd vs. 3rd column from the left). The MAP difference between 0.0754 of idf and 0.0798 of P(t | R) is not

Table 4. Retrieval performance of **P(*t* | *R*) guided bag of word expansion** on TREC 2007 Legal track, as measured by *MAP*/*statAP*. The baseline unexpanded queries produced an MAP/statAP of 0.0663/0.0160.

\ # query terms selected # expansions per query term	1	2	3	4	All
1	0.0755**/0.0210	0.0744 /0.0172	0.0808 /0.0271	0.0768 /0.0260	0.0764***/0.0260
2	0.0795**/0.0229	0.0814** /0.0216	0.0892** /0.0242	0.0883** /0.0243	0.0867***/0.0242
3	0.0789**/0.0217	0.0829***/0.0221	0.0880** /0.0206	0.0894** /0.0207	0.0878** /0.0206
4	0.0789**/0.0217	0.0821** /0.0205	0.0908***/0.0203	0.0993***/0.0213	0.0927***/0.0214
All	0.0791**/0.0219	0.0833** /0.0217	0.1006$^{\#}$ /0.0207	0.1038$^{\#}$ /0.0211	0.1014$^{\#}$ /0.0200

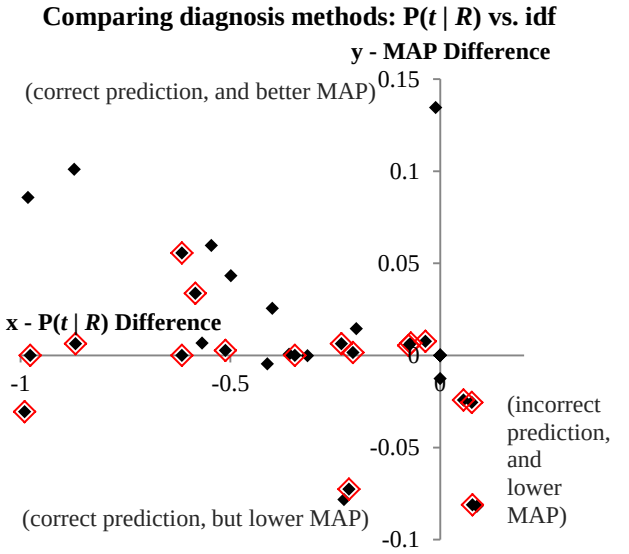

Figure 3. Difference in prediction accuracy vs. difference in MAP for the two selective query expansion methods on 43 TREC 2007 Legal Track topics. The X axis shows the difference in true P(*t* | *R*) between the first query terms selected by each method. The Y axis shows the difference in MAP between queries expanded by each method. The differences are calculated as that from predicted P(*t* | *R*) based diagnosis minus that from idf based diagnosis. Points surrounded by a diamond represent topics in which one method selected a term that had no expansions.

statistically significant. We investigate why below. According to the diagnostic expansion framework, two causes are possible, 1) the P(*t* | *R*) predictions are poor, selecting the wrong query terms to expand, or 2) the quality of the expansion terms that idf selected happen to be higher, causing the idf method to have better MAP sometimes, thus decreasing statistical significance.

To separate the effects of diagnosis and expansion, we plot the Legal 2007 topics along two dimensions in the scatter-plot Figure 3. The *x* axis represents the diagnosis dimension: the difference between the true P(*t* | *R*) of the two query terms selected by lowest predicted P(*t* | *R*) and highest idf. The *y* axis represents the expansion performance dimension: the difference between the Average Precision (AP) values of the two methods on a topic. When idf and predicted P(*t* | *R*) happen to select the same term to expand for a given topic, that topic would be plotted on the origin (*x*=0, *y*=0) – no difference in both diagnosis and expansion.

From Figure 3, firstly, most points have $x < 0$, meaning the P(*t* | *R*) predictions are better at finding the low P(*t* | *R*) terms than the idf based diagnosis. Secondly, most points are in the top left and bottom right quadrants, supporting the theory that expanding the term with lower P(*t* | *R*) leads to better retrieval performance. In the bottom right quadrant, occasionally idf method picks the right terms with lower P(*t* | *R*) to expand, and does better in retrieval.

However, there are three outliers in the bottom left quadrant, which are not fully explained by our theory. At the bottom left quadrant, predicted P(*t* | *R*) does identify the right term with a lower P(*t* | *R*), but the retrieval performance is actually worse than that of idf guided expansion.

By looking into these three topics, we found that the manual queries for topics 76 and 86 do not have any expansion terms for the query terms selected by P(*t* | *R*), while the idf selected terms do have effective expansion terms. All such topics where a query term without expansion terms is selected are annotated with diamond shaped borders in the plot. Topic 55 is because of poor expansion term quality. The P(*t* | *R*) method selects the chemical name *apatite* for expansion, which represent a class of chemicals. The manual expansion terms seem very reasonable, and are just names or chemical formulas of the chemicals belonging to the *apatite* family. However, the query is really about *apatite rocks* as they appear in nature, not any specific chemical in the apatite family. Thus, even expansion terms proposed by experts can still sometimes introduce false positives into the rank list, and this problem cannot be easily identified without corpus interaction, e.g. examining the result rank list of documents.

If these 3 topics were removed from evaluation, predicted P(*t* | *R*) guided expansion would be significantly better than idf guided expansion, at $p < 0.05$ by the two tailed sign test.

Of the 50 TREC 4 topics, similarly, 4 topics are outliers. In two cases, the P(*t* | *R*) selected query terms do not have manual expansion terms. In one topic, P(*t* | *R*) prediction did not select the right term, but MAP is higher than idf diagnosis, because the idf method selected a query term with poor expansion terms. In one topic, the retrieval performance does not differ at top ranks, and the idf method only excels after 30 documents.

Table 5. Retrieval performance of **P(*t* | *R*) guided bag of word expansion** on TREC 4 Ad hoc track, as measured by *MAP*. The baseline unexpanded queries produced an MAP of 0.1973.

\#qt #exp	1	2	3	4	All
1	0.2101**	0.2102*	0.2117**	0.2113**	0.2113**
2	0.2146***	0.2161***	0.2200**	0.2201**	0.2201**
3	0.2160***	0.2154***	0.2222***	0.2226***	0.2218**
4	0.2204$^{\#}$	0.2272***	0.2288***	0.2309**	0.2309**
All	0.2215$^{\#}$	0.2290**	0.2329**	0.2343**	0.2384**

Overall, most topics confirm our hypothesis that expanding the query term likely to mismatch relevant documents leads to better retrieval, and better diagnosis also leads to better retrieval.

This analysis also shows the inherent difficulty of evaluating term diagnosis in end-to-end retrieval experiments. Even with high quality manual expansion terms, there is still some variation in the quality of the expansion interventions, which can still interfere with the assessment of the diagnosis component.

5.5 Boolean CNF vs. bag of word expansion

We compare CNF style expansion with *two* advanced bag-of-word expansion methods.

For a fair comparison with manual CNF expansion, our *first* bag of word expansion baseline also uses the set of manual expansion terms selected by predicted P(*t* | *R*). Expansion terms are then grouped and combined with the original query for retrieval.

To make this baseline strong, both individual expansion terms and the expansion term set can be weighted. The individual expansion terms are weighted with the Relevance Model weights [17] from an initial keyword retrieval, with the parameter (the number of feedback documents) tuned on the training set. Manual expansion terms that do not appear in the feedback documents are still included in the final query, but a minimum weight is used to conform to the relevance model weights. Uniform weighting of the expansion terms was also tried. It is more effective than relevance model weights when expansion is more balanced, i.e. more than 3 query terms are selected for expansion. When combining the expansion terms with the original query, the combination weights are 2-fold cross-validated on the test set.

Table 4 shows *the best case* of both relevance-model-weight and uniform-weight bag of word expansion. Bag of word expansion performs worse than CNF expansion in almost all the different setups. The best performance is achieved with full expansion of 4 query terms, with a MAP of 0.1038, slightly lower than that of CNF (0.1039 in Table 1), however, the statAP value of 0.0211 is much worse than that of CNF (0.0508, Table 2).

Table 5 shows the best case bag of word expansion results on the TREC 4 Ad hoc dataset. Consistent with the statAP measure on TREC Legal 2007, CNF queries are much better than bag of word expansion. For example, with full expansion of all query terms, CNF expansion (Table 3) gets a MAP of 0.2938, 23% better than 0.2384 of the bag of word expansion with the same expansion terms, significant at $p < 0.0025$ by the randomization test and weakly significant at $p < 0.0676$ by the sign test.

Some results of bag of word retrieval at low selection levels, i.e. selecting one query term to expand, perform better than idf guided CNF expansion. But since the bag of word expansion here uses better expansion terms selected by predicted P(*t* | *R*), this does not mean that bag of word is sometimes better than CNF expansion.

The *second* bag of word expansion baseline is the standard Lavrenko Relevance Model itself [17], which uses automatic feedback terms, instead of manual ones, for expansion. Parameters trained on Legal 2006 dataset when applied to Legal 2007 lead to an MAP of 0.0606, statAP of 0.0168, worse than the no expansion baseline. On TREC 4, it gets a MAP of 0.2488, slightly better than 0.2384, the best manual bag of word expansion, but still much worse than CNF (0.2938).

In sum, given the same set of high quality expansion terms, CNF expansion works much better than bag of word expansion.

6. CONCLUSIONS

We set out with the hypothesis that term mismatch based diagnosis can successfully guide further retrieval intervention to fix 'problem' terms and improve retrieval. In this work, we applied the term mismatch diagnosis to guide interactive query expansion. Simulated interactive query expansion experiments on TREC Ad hoc and Legal track datasets not only confirmed this hypothesis, but also showed that *automatically predicted* P(*t* | *R*) probabilities (the complement of term mismatch) can accurately guide expansion to the terms that need expansion most, and lead to better retrieval than when expanding rare terms first. From the user's point of view, it usually isn't necessary to expand every query term. Guided by predicted P(*t* | *R*), expanding two terms is enough for most topics to achieve close-to-top performance, while guided by idf (rareness), three terms need to be expanded. P(*t* | *R*) guidance can save user effort by 33%.

In addition to confirming the main hypothesis, experiments also showed that Boolean conjunctive normal form (CNF) expansion outperforms carefully weighted bag of word expansion, given the same set of high quality expansion terms. The unstructured bag of word expansion typically needs balanced expansion of most query terms to achieve a reliable performance.

Although the effect from adding more expansion terms to a query term diminishes, for the query terms that do need expansion, the effects of the expansion terms are typically additive, the more the expansion the better the performance. This is consistent with prior observations on vocabulary mismatch, that even after including more than 15 aliases, the effects of mismatch can still be observed, and further expansion may still help [7].

For bag of word expansion, including more manual expansion terms also helps, but requires a balanced expansion of most query terms, and is not as effective and stable as CNF expansion.

This work is mostly concerned with automatic diagnosis of problem terms in the query, and presents them to the user for manual expansion. It is still a question whether the diagnosis can help automatic formulation of effective CNF queries. We hope to let the system suggest or select expansion terms, automatically or semi-automatically with minimal user effort. Automatic identification of high quality expansion terms would be useful when the candidate expansion terms may not be of high quality, e.g. expansion terms from result documents, thesaurus or non-expert users. Poor expansion terms in CNF queries are especially harmful, when they over-generalize the query and introduce false positives throughout the rank list. Tools such as performance prediction methods (e.g. query clarity), may help in such scenarios to detect the adverse effects of the poor expansion terms.

In the future, we also hope to diagnose precision related problems as well as mismatch problems. We can then use the diagnosis to guide disambiguation, phrasing or fielded retrieval, as well as term substitution, removal or expansion.

7. ACKNOWLEDGEMENTS

This work is supported by NSF grant IIS-1018317. Opinions in this work are solely the authors'. We thank Chengtao Wen, Grace Hui Yang, Jin Young Kim, Charlie Clarke, Gordon Cormack and NIST for helpful discussions, feedback and access to data.

8. REFERENCES

[1] J. A. Aslam and V. Pavlu. A practical sampling strategy for efficient retrieval evaluation. Report. May 2007.

[2] J. R. Baron, D. D. Lewis and D. W. Oard. TREC 2006 Legal Track Overview. In *Proceedings of the fifteenth Text REtrieval Conference (TREC '06)*, 2007.

[3] G. Cao, S. Robertson and J. Nie. Selecting Query Term Alterations for Web Search by Exploiting Query Contexts. In

Proceedings of 46th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies (ACL-08: HLT). 148-155, 2008.

[4] C. L. A. Clarke, G. V. Cormack and F. J. Burkowski. Shortest Substring Ranking (MultiText Experiments for TREC-4). In *Proceedings of the Fourth Text REtrieval Conference (TREC-4)*. 1996.

[5] V. Dang and W. B. Croft. Query reformulation using anchor text. In *Proceedings of the 3rd ACM International Conference on Web Search and Data Mining (WSDM '10*). 41-50, 2010.

[6] S. Deerwester, S. T. Dumais, G. W. Furnas, T. K. Landauer and R. Harshman. Indexing by latent semantic analysis. *Journal of the American Society for Information Science*, 41(6): 391-407. 1990.

[7] G. W. Furnas, T. K. Landauer, L. M. Gomez, and S. T. Dumais. The vocabulary problem in human-system communication. *Communications of ACM*, 30(11): 964-971. ACM. New York, NY. November, 1987.

[8] W. Greiff. A theory of term weighting based on exploratory data analysis. In *Proceedings of 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '98)*. 11-19, 1998.

[9] D. Harman. Towards interactive query expansion. In *Proceedings of the 11th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '88)*. 321-331, 1988.

[10] D. Harman. Overview of the Third Text REtrieval Conference (TREC-3). In *Proceedings of the 3rd Text REtrieval Conference (TREC '94)*, 1995.

[11] D. Harman. Overview of the Third Text REtrieval Conference (TREC-4). In *Proceedings of the 4th Text REtrieval Conference (TREC '95)*, 1996.

[12] S. Harter. *Online Information Retrieval: Concepts, Principles, and Techniques*. Academic Press. San Diego, California. 1986.

[13] M. Hearst. Improving full-text precision on short queries using simple constraints. In *Proceedings of the Fifth Annual Symposium on Document Analysis and Information Retrieval (SDAIR '96)*, 1996.

[14] R. Jones, B. Rey, O. Madani and W. Greiner. Generating query substitutions. In *Proceedings of the 15th International Conference on World Wide Web (WWW '06)*. 387-396, 2006.

[15] J. Lamping and S. Baker. Determining query term synonyms within query context. United States Patent No. 7,636,714. USPTO March, 2005.

[16] W. Lancaster. *Information Retrieval Systems: Characteristics, Testing and Evaluation*. Wiley. New York, New York, USA. 1968.

[17] V. Lavrenko and W. B. Croft. Relevance-based language models. In *Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '01)*. 120-127, 2001.

[18] J. Lin and M. D. Smucker. How do users find things with PubMed?: towards automatic utility evaluation with user simulations. In *Proceedings of the 31st annual international ACM SIGIR conference on Research and Development in Information Retrieval (SIGIR '08)*. 19-26, 2008.

[19] D. Metzler and W.B. Croft. Combining the language model and inference network approaches to retrieval. *Information Processing and Management Special Issue on Bayesian Networks and Information Retrieval*, 40(5), 735-750, 2004.

[20] D. Metzler. Generalized inverse document frequency. In *Proceedings of the 17th ACM Conference on Information and Knowledge Management (CIKM '08)*. 399-408, 2008.

[21] M. Mitra, A. Singhal and C. Buckley. Improving automatic query expansion, *Proceedings of 21st annual international ACM SIGIR conference on Research and Development in Information Retrieval (SIGIR '98)*. 206-214, 1998.

[22] F. Peng, N. Ahmed, X. Li and Y. Lu. Context sensitive stemming for Web search. In *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '07)*. 639-646, 2007.

[23] S. E. Robertson and K. Spärck Jones. Relevance weighting of search terms. *Journal of the American Society for Information Science*, 27(3):129-146. 1976.

[24] G. Salton, E. A. Fox and H. Wu. Extended Boolean information retrieval. *Communications of ACM*, 26(11): 1022-1036. ACM. New York, NY. November 1983.

[25] S. Tomlinson. Experiments with the Negotiated Boolean Queries of the TREC 2006 Legal Discovery Track. In *Proceedings of the fifteenth Text REtrieval Conference (TREC '06)*, 2007.

[26] S. Tomlinson, D. W. Oard, J. R. Baron and P. Thompson. Overview of the TREC 2007 Legal Track. In *Proceedings of the sixteenth Text REtrieval Conference (TREC '07)*, 2008.

[27] E. Tudhope. Query based stemming. PhD Thesis. University of Waterloo. 1996.

[28] X. Wang and C. Zhai. Mining term association patterns from search logs for effective query reformulation. In *Proceedings of the 17th ACM Conference on Information and Knowledge Management (CIKM '08)*. 479-488, 2008.

[29] R. W. White, I. Ruthven, J. M. Jose, and C. J. Van Rijsbergen. Evaluating implicit feedback models using searcher simulations. *ACM Trans. Inf. Syst.* 23(3): 325-361. July, 2005.

[30] X. Xue, W. B. Croft and D. A. Smith. Modeling reformulation using passage analysis. In *Proceedings of the 19th ACM Conference on Information and Knowledge Management (CIKM '10)*. 2010.

[31] L. Zhao and J. Callan. Effective and efficient structured retrieval (poster description). In *Proceedings of the 18th ACM Conference on Information and Knowledge Management (CIKM '09)*. 1573-1576, 2009.

[32] L. Zhao and J. Callan. Term necessity prediction. In *Proceedings of the 19th ACM Conference on Information and Knowledge Management (CIKM '10)*. 2010.

[33] Y. Zhu, L. Zhao, J. Callan and J. Carbonell. Structured queries for legal search. In *Proceedings of the sixteenth Text REtrieval Conference (TREC '07)*, 2008